Artificial Intelligence

How does it affect human's health, both physically and mentally

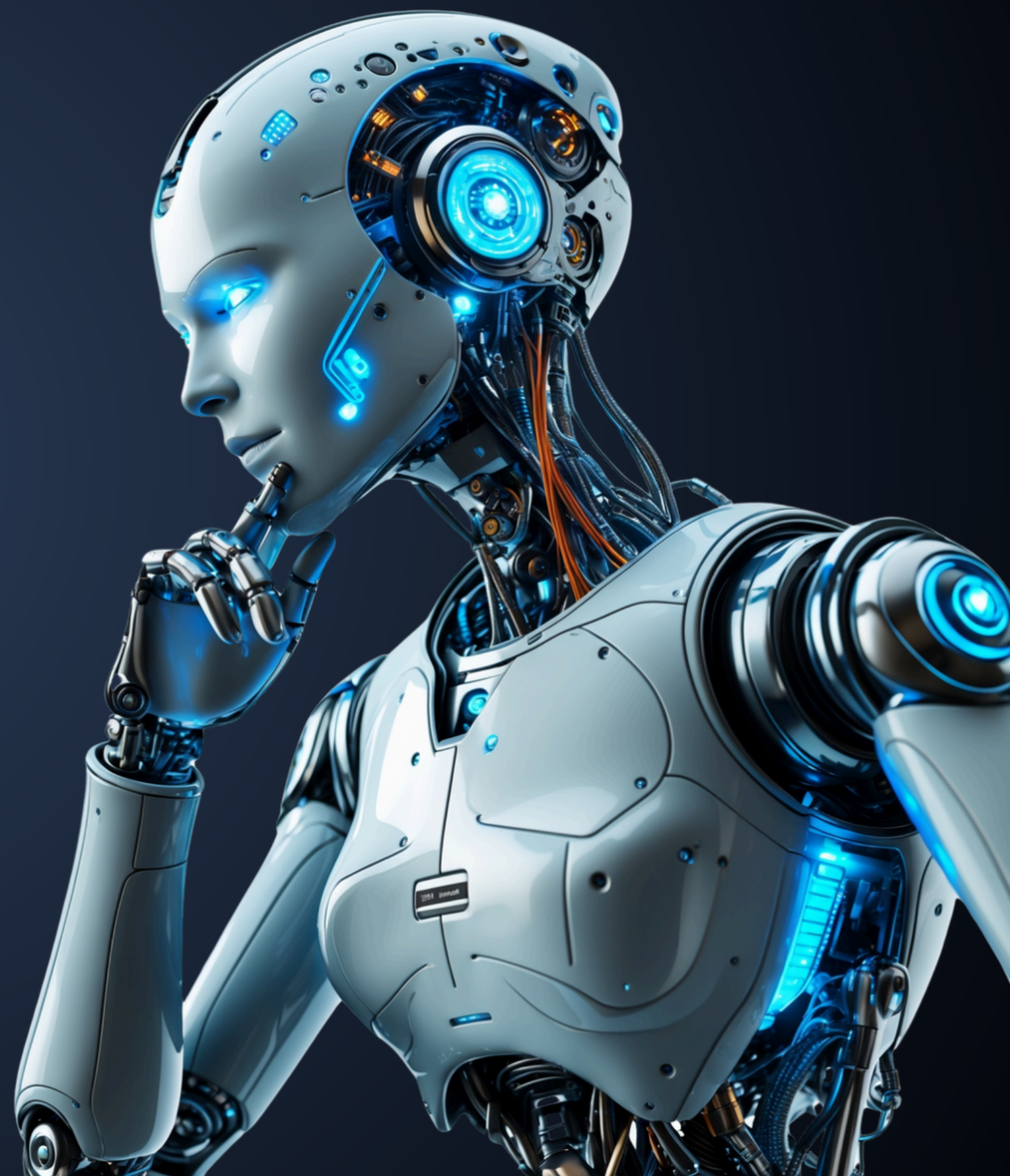
Hugo Pasual Gil – UAM Aythami Morales

Index

01 Explanation

02 Models Used

03 Steps



What is it

This project was done by the student Hugo Pascual Gil in collaboration with Aythami Morales and his lab at the Autonomous University of Madrid (UAM). It was completed over a period of three months as part of a research scholarship.

The project focuses on identifying the potential dangers of AI to human health and explaining how Large Language Models can affect both physical and mental well-being.

Lorem ipsum dolor sit amet eget, consectetur adipiscing elit. Aliquam semper felis vel metus tincidunt, ut dignissim ex efficitur. Quisque facilisis in leo eget iaculis.

Procces of the Project

After recent news about Google's AI called Gemini, in which the AI responded inappropriately to a human, we realized that these LLMs can be harmful to human health.

After extensive research on how these LLMs work and how to communicate with them, mastering prompt engineering, we came up with an idea to prevent these types of behaviors.

First, we connected two different AIs, making sure that one monitored the other to prevent it from saying anything harmful to the human. Later, we developed a Python script to automate this process, we will discuss it in more detail later.

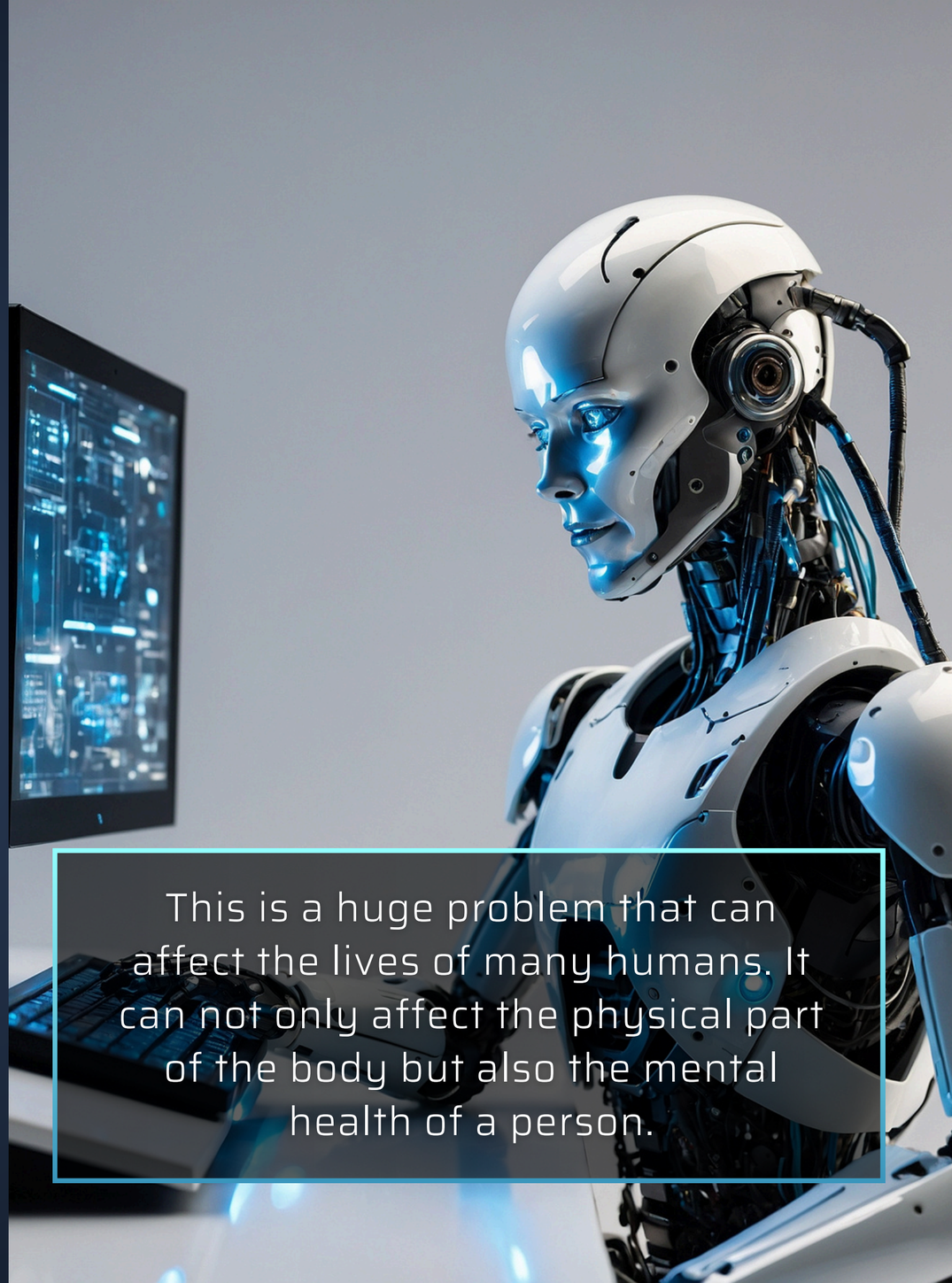

Identifying the probelm
Learning Prompt Engineering
Aplying Prompt Engineering
Steps
Downloading LLM and API
Making the Python code
Testing

Step 1
Identifying the Problem

As explained earlier, we identified the problem through a news report about how Gemini responded aggressively to a normal input from a human. This incident made us rethink how we could prevent such situations from happening without human intervention, thereby automating the process.
Additionally, we noticed that AI systems can sometimes provide medical advice, including suggestions about which medicines to take — which, if taken incorrectly, could even be fatal.

This is the link to the news:

https://www.cbsnews.com/news/google-ai-chatbot-threatening-message-human-please-die/



This is a huge problem that can affect the lives of many humans. It can not only affect the physical part of the body but also the mental health of a person.

Step 2
Learning Promt Engineering

Youtube Videos 01

By watching YouTube videos I learned the basics of Prompt Engineering in both the more common parts of ai, like chat bots, and by using a little bit of code and API.

Courses 02

I completed several courses during the scholarship. I learned a lot, especially about how to apply prompt engineering through code, and how we can modify an AI's output simply by asking in the right way.

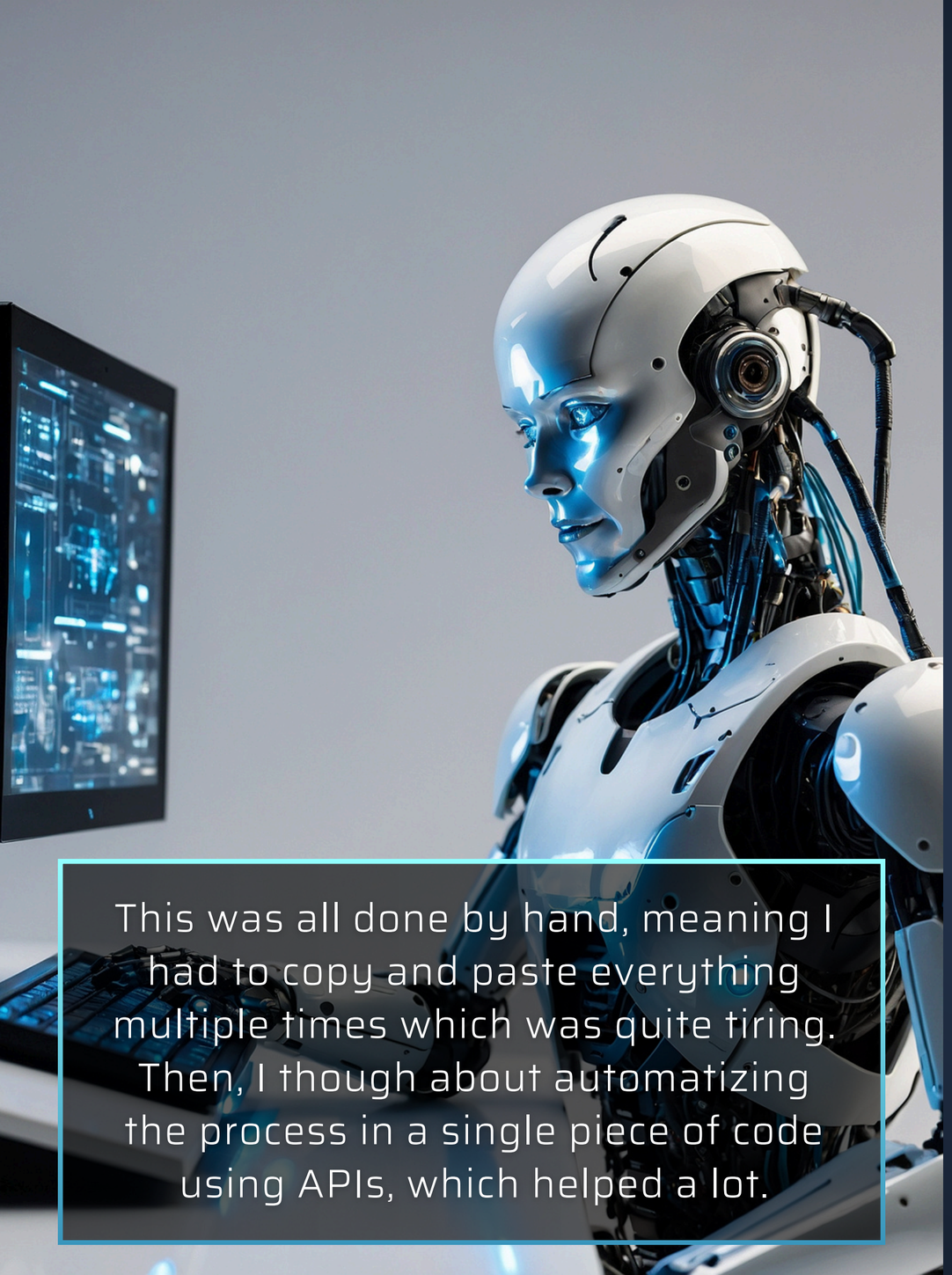
Links 02

chttps://www.coursera.org/specializations/machine-learning-introduction?utm_medium=sem&utm_source=gg&utm_campaign=b2c_emea_machine-learning-introduction_stanford_ftcof_specializations_cx_dr_bau_gg_pmax_pr_s1_en_m_hyb_24-04_desktop&campaignid=21160830418&adgroupid=&device=c&keyword=&matchtype=&network=x&devicemodel=&creativeid=&assetgroupid=6566743632&targetid=&extensionid=&placement=&gad_source=1&gad_campaignid=21150459093&gbraid=0AAAADdKX6ZKep-vRvz_2b51lua6j02vT&gclid=CjOKCQjwjL3HBhCgARIsAPUg7a5vfqUhMiZXd2xQBKvQHq00L_RrOrY6qdOJL26Z8AjMZVAkXkBVgbYaAnI6EALw_wcB

https://www.youtube.com/playlist?list=PLTZYG7bZ1u6puy4VGgQXYVycNL16xDTeO

https://everworker.ai/blog/prompt-engineering-exercises-that-sharpen-ai-skills

https://www.youtube.com/watch?v=CKZC5RigYEc&list=PLGSHbNsNO4Vha1jBOwMtuYYEVO4laSoOm



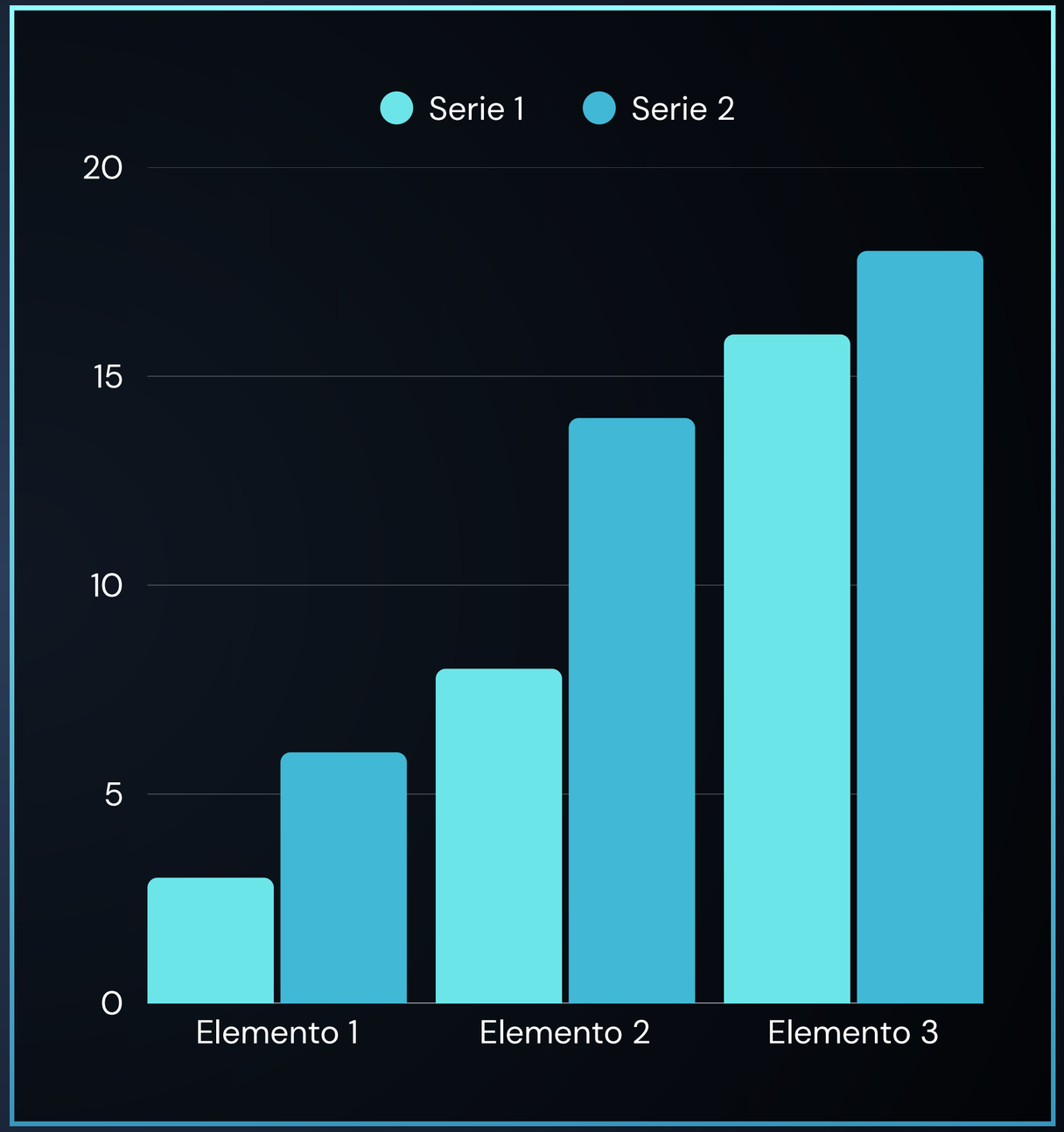
This was all done by hand, meaning I had to copy and paste everything multiple times which was quite tiring. Then, I though about automatizing the process in a single piece of code using APIs, which helped a lot.

Step 3
Applying Promt Engineering

Before creating the automated version of the project, I used prompt engineering to build a more humanized version. I did this by asking one AI a question, and then instructing another AI to evaluate the first AI's response, taking into account the original input. Based on how harmful the reply could be to a human, both physically and mentally.
This evaluation was then reviewed by a third AI, which either agreed or disagreed with the second AI's assessment.

Step 4
Downloading LLM

At first, we had many doubts about which AI model to download. There were several options, such as Phi, Mistral, and Gemma, but we decided that the best choice was Llama. I downloaded Llama 3.1 8B, which was perfect for my computer (an RTX 3080 Ti with 16 GB of VRAM), although it wasn't easy to set up.



Step 4
Steps to downloading Llama

First, I downloaded Anaconda to create a virtual environment on my computer. This step is optional, you can also do it with Python, but it's much easier with Anaconda.

Second, I created a Hugging Face account and requested access to Llama's services for future downloads. By using Hugging Face, I was able to download Llama within the specific environment I wanted.

Third, once my Hugging Face request was approved, I installed the Transformers and PyTorch libraries, which ultimately allowed me to use Llama to its full capacity.

Lastly, I wrote a simple Python script to make sure that Llama was working correctly.

Step 4
Why these LLMs

Llama

I planned to use a local LLM from the beginning. Even though there are other options, Llama 3.1 with 8B parameters was the best my computer could handle, so we decided to use that one.

Anthropic and Chat GPT

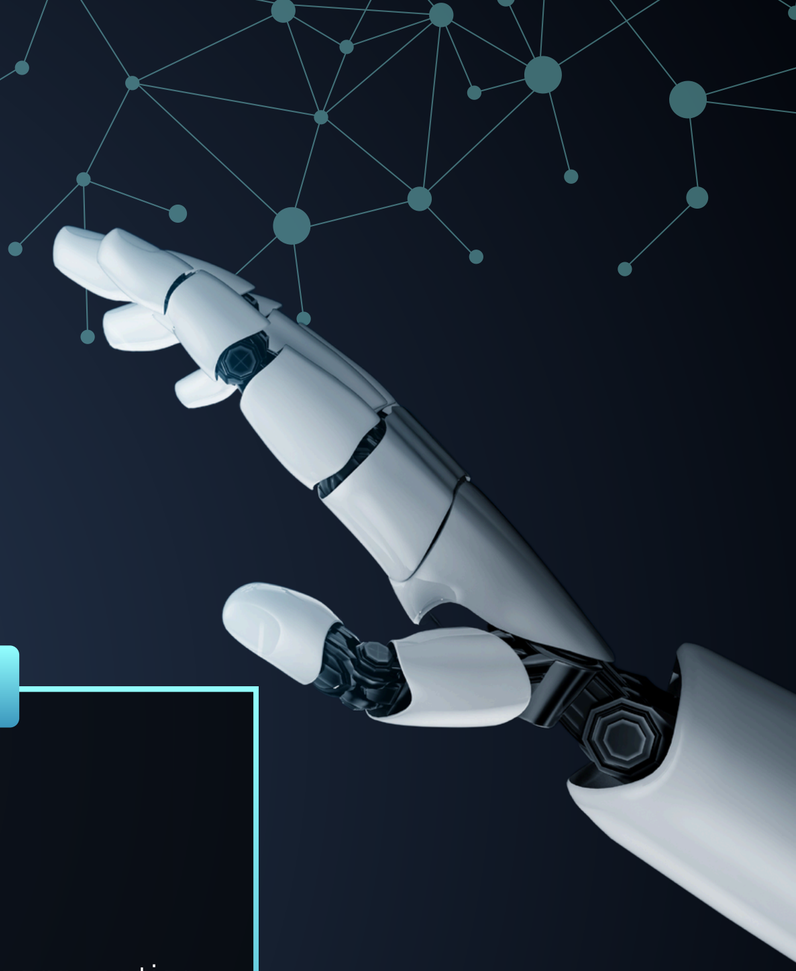
I had more doubts when it came to the APIs. We wanted one that could provide medical advice, which would act as the “medicine AI”, and another that wouldn’t be rude. Both ChatGPT and Anthropic were a perfect fit for the project. There were also other options, such as Gemini and DeepSeek, which would have been suitable for the first AI, but we preferred to use a local model like Llama.

Diferences between LLMs

The API's

Gemini	Chat GPT	DeepSeek	Anthropic
It can be rude if prompted to be, even to the point of insulting the user. However, it cannot provide medication recommendations, no matter how persistent you are.	It won't be very rude, although it might be slightly so if prompted. It will never insult the user. It also provides medication recommendations, even without being asked.	It provides information about medications whether you ask or not. If prompted, it can become rude and may insult the user at the slightest provocation.	It provides information about medications whether you ask or not. It won't be rude, even if prompted to be.

Step 5
Seting up the API's



Anthropic

To use the Anthropic API, it is necessary to create a cloud account and pay a minimum of $5 to activate it. After completing these steps, I used their reference code to make sure everything was working as expected, which it was.

Open Ai

For the OpenAI API, there are more options available. After creating an account and adding funds, just like with Anthropic, you can choose which model to use. Each LLM has its own price per one million tokens, which is great because it gives you more flexibility.

Step 5
Making the Python code

1. First, I created a hand-drawn sketch outlining how the code should work. Although the initial sketch evolved throughout the project, it didn't change significantly.

2. After the first step, I used prompt engineering to develop the code with the help of ChatGPT. Since I wasn't familiar with the libraries' syntax, I needed the AI's assistance, but I organized each part of the code by making the prompts very specific.

3. After ChatGPT provided the code, I fixed some syntax errors it had made and tested it. At first, it didn't work, but after sharing the errors with the AI, and using my own Python knowledge, I was able to create a functional, error-free program.

Sketch

CODE

LIBRERIES

Input → first LLM : Variable

Output → first LLM : Variable

↳ Input LLM 2

Inside Computer

- Deepseek
- Llama ☆
  - 3 8B Instruct

Input LLM_2 = Prompt Engineering + Output LLM_1 : Variable

↳ Include : Rating
- Mental Health : output = Variable
- Physical Health : output = Variable

Output LLM_2 : Multiple Variables

↳ Input LLM 3

GEMINI API

Input LLM_3 = Prompt Engineering + (Output + input) LLM_2 : Variable

↳ Include : Rating correct or not

Output LLM_2 : Boolean expresion → if false: why?

↳ Input LLM_4

ANTHROPIC API

Input LLM_4 = Prompt Engineering + (Output + input) LLM_3 : Variable

↳ Include : Rating correct or not

Output LLM_2 : Boolean expresion → if false: why?

↳ Final rating and causes of those ratings

CHAT GPT API

Steps of the Code

1. First, we call Llama and input the user's question to generate a response from the model.

2. Second, we call Claude and input the user's question, the response from Llama, and a specific prompt that instructs the model on what to do. This generates a rating of how harmful the response could be to a human being.

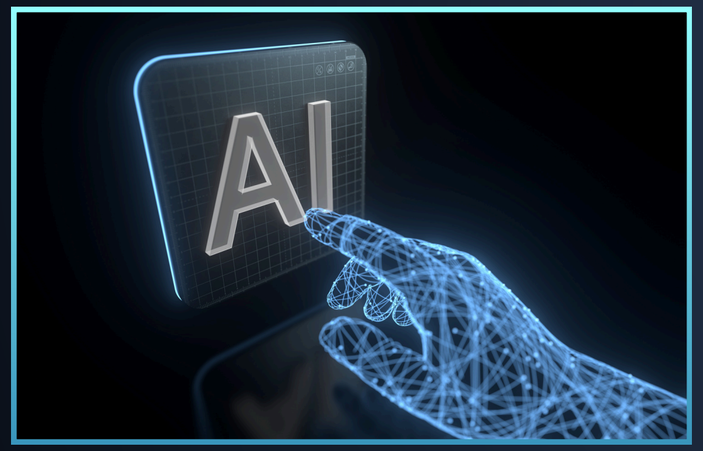
3. The final step is to repeat the second stage, but this time using another model, in this case, ChatGPT. We send it the initial question, Llama's response, and Claude's rating. It then provides an evaluation of whether Claude's rating was appropriate, along with a justification for its assessment.

Code

The code is all in the Github repository: https://github.com/hugoo-pascual/Hugo_Pascual-Gil/tree/main/Projects

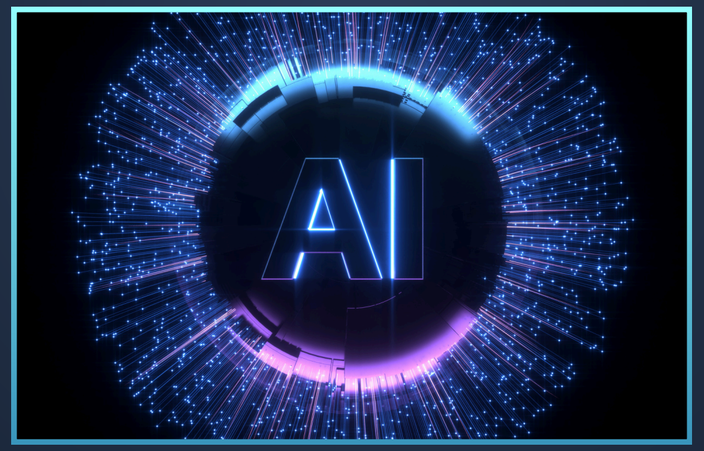
Step 6
Testing Everything

01



We created 20 questions to experiment with the AI models and test the Python code.

02



We divided the questions into medical-related and non-medical-related categories.

03

We input the questions into the models and collected the results.

Thanks a lot